

(19) 日本国特許庁 (J P)

(12) 特 許 公 報 (B 2)

(11) 特許番号

特許第3083805号
(P3083805)

(45) 発行日 平成12年 9 月 4 日 (2000. 9. 4)

(24) 登録日 平成12年 6 月 30 日 (2000. 6. 30)

(51) Int.Cl.⁷

識別記号

F I

G 0 6 F 17/30
11/34

G 0 6 F 15/40 3 7 0 G
11/34 C
15/419 3 2 0
15/401 3 3 0 Z
15/403 3 4 0 B

請求項の数14(全 14 頁)

(21) 出願番号 特願平10-243854
(22) 出願日 平成10年 8 月 28 日 (1998. 8. 28)
(65) 公開番号 特開2000-76266(P2000-76266A)
(43) 公開日 平成12年 3 月 14 日 (2000. 3. 14)
審査請求日 平成11年 7 月 22 日 (1999. 7. 22)

(73) 特許権者 390009531
インターナショナル・ビジネス・マシー
ンズ・コーポレーション
INTERNATIONAL BUSI
NESS MACHINES COR
PORATION
アメリカ合衆国10504、ニューヨーク州
アーモンク (番地なし)
(72) 発明者 青木 義則
神奈川県大和市下鶴間1623番地14 日本
アイ・ピー・エム株式会社 東京基礎研
究所内
(74) 代理人 100086243
弁理士 坂口 博 (外 1 名)

審査官 高瀬 勤

最終頁に続く

(54) 【発明の名称】 ブラウザ操作自動実行システムおよびその方法

(57) 【特許請求の範囲】

【請求項 1】 ブラウザ操作を記録する、操作イベント記
録システムであって、

(1) イベント記録プログラムを読み込むブラウザを起
動する手段と、

(2) コンテンツを表示するブラウザを起動する手段
と、

(3) 前記コンテンツを表示するブラウザに、イベント
記録に必要な情報を収集するイベントハンドラの埋め込
まれたページを読み込む手段と、

(4) 前記コンテンツを表示するブラウザにおけるユー
ザのブラウザ操作に応じて、前記イベントハンドラに従
い、イベントを発生させる手段と、

(5) 前記イベント記録プログラムにより前記発生した
イベントを記録する手段と、

を具備する、操作イベント記録システム。

【請求項 2】 ブラウザ操作を自動実行する、自動実行シ
ステムであって、

(1) ブラウザにイベント記録／再生プログラムを読み
込む手段と、

(2) コンテンツを表示するブラウザを起動する手段
と、

(3) 前記コンテンツを表示するブラウザに、イベント
記録または再生に必要な情報を収集するイベントハンド
ラの埋め込まれたページを読み込む手段と、

(4) 前記コンテンツを表示するブラウザにおけるユー
ザのブラウザ操作に応じて、前記イベントハンドラに従
い、イベントを発生させる手段と、

(5) 前記イベント記録／再生プログラムにより前記発
生したイベントを記録する手段と、

(6) 記録したイベントを前記イベント記録／再生プログラムにより読み込む手段と、

(7) 前記読み込んだイベントに応じて、前記コンテンツを表示するブラウザを動作させる手段と、
を具備することを特徴とする、自動実行システム。

【請求項3】前記イベントを記録する手段(5)が、URLの遷移、ポインタの移動、ウェブのコンテンツが提供するテキストフィールドまたはラジオボタンまたは選択リストなどのフォームへの操作、前記コンテンツを表示するブラウザへのウィンドウ操作を含む、操作を記録する手段である、請求項1または2の何れかに記載のシステム。

【請求項4】前記イベントを記録する手段(5)が、前記コンテンツを表示するブラウザへの特定の操作のイベントのみを記録する手段である、請求項1または2の何れかに記載のシステム。

【請求項5】前記イベントを記録する手段(5)が、前記コンテンツを表示するブラウザへの文字または図形の描画操作のイベントを記録する手段を含む、請求項1または2の何れかに記載のシステム。

【請求項6】前記コンテンツを表示するブラウザを動作させる手段(7)が、可変速再生または特定の操作のみの再生を行う手段を含む、請求項2記載のシステム。

【請求項7】前記イベントを記録する手段(5)が、前記発生したイベントをテキストファイルに記録することを特徴とする、請求項1または2の何れかに記載のシステム。

【請求項8】前記コンテンツを表示するブラウザを動作させる手段(7)が、前記テキストファイルを編集して、ブラウザの再生動作を変化させる手段を含む、請求項7記載のシステム。

【請求項9】ブラウザ操作を記録する、操作イベント記録方法であって、

(1) イベント記録プログラムを読み込むブラウザを起動し、コンテンツを表示するブラウザを起動する段階と、

(2) 前記コンテンツを表示するブラウザに、イベント記録に必要な情報を収集するイベントハンドラの埋め込まれたページを読み込む段階と、

(3) 前記コンテンツを表示するブラウザにおけるユーザのブラウザ操作に応じて、前記イベントハンドラに従い、イベントを発生させる段階と、

(4) 前記イベント記録プログラムにより前記発生したイベントを記録する段階と、
を有することを特徴とする、操作イベント記録方法。

【請求項10】前記ブラウザを起動する段階(1)が、コンテンツを表示するブラウザを起動し、イベント記録プログラムを読み込むブラウザを起動する段階である、請求項9記載の方法。

【請求項11】ブラウザ操作を自動実行する、自動実行

方法であって、

(1) ブラウザにイベント記録／再生プログラムを読み込み、コンテンツを表示するブラウザを起動する段階と、

(2) 前記コンテンツを表示するブラウザに、イベント記録または再生に必要な情報を収集するイベントハンドラの埋め込まれたページを読み込む段階と、

(3) 前記コンテンツを表示するブラウザにおけるユーザのブラウザ操作に応じて、前記イベントハンドラに従い、イベントを発生させる段階と、

(4) 前記イベント記録／再生プログラムにより前記発生したイベントを記録する段階と、

(5) 記録したイベントを前記イベント記録／再生プログラムにより読み込む段階と、

(6) 前記読み込んだイベントに応じて、前記コンテンツを表示するブラウザを動作させる段階と、
を有することを特徴とする、自動実行方法。

【請求項12】前記ブラウザを起動する段階(1)が、コンテンツを表示するブラウザを起動し、イベント記録／再生プログラムを読み込むブラウザを起動する段階である、請求項11記載の方法。

【請求項13】ブラウザ操作を記録するためのプログラムを記録したコンピュータ読み取り可能な記録媒体であって、該プログラムがコンピュータに、

(1) イベント記録プログラムを読み込むブラウザを起動し、コンテンツを表示するブラウザを起動する機能と、

(2) 前記コンテンツを表示するブラウザに、イベント記録に必要な情報を収集するイベントハンドラの埋め込まれたページを読み込む機能と、

(3) 前記コンテンツを表示するブラウザにおけるユーザのブラウザ操作に応じて、前記イベントハンドラに従い、イベントを発生させる機能と、

(4) 前記イベント記録プログラムにより前記発生したイベントを記録する機能、
を実現させる、コンピュータ読み取り可能な記録媒体。

【請求項14】ブラウザ操作を自動実行するためのプログラムを記録したコンピュータ読み取り可能な記録媒体であって、該プログラムがコンピュータに、

(1) ブラウザにイベント記録／再生プログラムを読み込み、コンテンツを表示するブラウザを起動する機能と、

(2) 前記コンテンツを表示するブラウザに、イベント記録または再生に必要な情報を収集するイベントハンドラの埋め込まれたページを読み込む機能と、

(3) 前記コンテンツを表示するブラウザにおけるユーザのブラウザ操作に応じて、前記イベントハンドラに従い、イベントを発生させる機能と、

(4) 前記イベント記録／再生プログラムにより前記発生したイベントを記録する機能と、

(5) 記録したイベントを前記イベント記録／再生プログラムにより読み込む機能、

(6) 前記読み込んだイベントに応じて、前記コンテンツを表示するブラウザを動作させる機能と、を実現させる、コンピュータ読み取り可能な記録媒体。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本願は、ウェブブラウザ(Web Browser)操作自動実行システムおよびその方法に関し、特に既存のウェブブラウザへの操作イベントを記録／編集／再生するプログラムによって記録した操作イベントをウェブブラウザで再生する、自動実行システムおよびその方法に関する。なお、本願明細書において、ウェブブラウザとはネットスケープ社のネットスケープコミュニケーター(Netscape Communicator)、マイクロソフト社のインターネットエクスプローラ(Internet Explorer)などの典型的なウェブ上のコンテンツを見るためのブラウザを指すものとする。

【0002】

【従来の技術】従来、ウェブ上で操作の記録／再生を行う技術として、以下のものが知られている。

(1) SMIL (Synchronized Multimedia Integration Language)

XML1.0を適用した記述言語で、動画、音声、図形、テキストなど異なるメディアのオブジェクトを同期させることができる。SMILを使用することで、ウェブ上でスライド、音声、ビデオ映像、文書を同期させたマルチメディア・プレゼンテーションを実現することができる。リアルシステムG2 (RealNetworks社のRealSystem G2) は、SMILに対応したストリーミング・メディア・プレイヤーである。しかしながらこの技術では、コンテンツをSMILで記述する必要があり、既存のHTML (Hyper Text Markup Language) コンテンツがそのままでは使えない。またコンテンツを表示するウィンドウのサイズ変更、位置変更などのウィンドウ操作、およびマウスの移動などを扱えない。さらにコンテンツを表示するウィンドウ上への任意の図形や文字の表示機能が無い。

(2) サイトクルーズ／パーソナル (SiteCruise/Personal)

NEC社によるウェブページ自動閲覧ソフトで、URL (Uniform Resource Locator) 遷移とスクロールの自動実行を行なう。記録したデータは“シナリオ”と呼ばれ、専用のエディタによって作成、編集される。このソフトウェアでは、ボタン、テキストフィールド、ラジオ・ボタンなどのフォームに対する操作、ウィンドウ操作、およびマウスの動きの記録・再生はできない。また専用のエディタを用いて“シナリオ”を書く必要があるため、専用エディタの操作に習熟する必要がある。その他、事前にソフトウェアをクライアントにインストールする必要があること、図形の描画機能が無いなどの欠点がある。

(3) VDOLive3.0

画像にあわせてウェブブラウザにコンテンツをブッシュ型配信するソフトウェアである。映像に同期してコンテンツを読み込む機能を実現している。このソフトウェアでは、ブラウザ操作を記録してシナリオを記述する機能が無い点、コンテンツを表示するウィンドウへのウィンドウ操作やフォームへの操作を記録・再生する機能が無い点、図形の描画機能が無い点。また、事前にVDOLive 3.0をシステムにインストールしておく必要がある。

(4) ロータス・スクリーンカム

画面イメージを、そのままビデオのように記録するソフトウェア。このソフトウェアは画面イメージをそのまま記録するためデータ・サイズが肥大化する。また、実際にアプリケーションを起動しているわけではないため、再生を一時停止し、途中からユーザが操作の続きを行なうといった使い方ができない。図形の描画機能も無い。記録したデータの編集ができない。また、事前にロータス・スクリーンカムをシステムにインストールしておく必要がある。

(5) マイクロソフト カムコーダー (Microsoft Camcorder)

画面イメージをそのままビデオのように記録・再生するソフトウェア。特徴は、ロータス・スクリーンカムと同じである。

(6) ビジュアルエイジテスト (VisualAge Test)

IBM社製のデバッグ・ツールで、GUIアプリケーションのテストを支援するプログラムである。ビジュアルエイジテストにはレコーダーとプレーヤーの機能があり、レコーダーとプレーヤーはユーザのキー操作とマウス操作を記録し、再生する。操作の記録はレコーダーによりロータススクリプト言語で記録され、スクリプト・ファイルに保管される。プレーヤーは、そのスクリプト・ファイルに従ってユーザーの操作を再現できる。このプログラムはウェブブラウザに対する操作のうち、いくつかの情報を記録できない。例えば、HTML中に記述されたアンカーをクリックする操作を記録できないため、URLの遷移を記録できない。これは、ビジュアルエイジテストのレコーダーで記録できる情報が、OSが提供する標準のGUIコンポーネントに対する操作のみに限定されるからである。また、ビジュアルエイジテストのプレーヤーを用いた再生では、コンテンツの読み込み終了を確認しないため、再生時にネットワーク環境などによってコンテンツの読み込みが遅くなった場合などに対応できない。更に、図形の描画機能が無い。可変速再生や特定の操作のみを再生するなどの特殊記録・再生ができない。

【0003】以上のように、従来技術では、既存のHTMLコンテンツをそのまま使用して、ウェブブラウザに対する全ての操作を記録・再生する方法を提供しない。

【0004】

【発明が解決しようとする課題】従って、本発明が解決

しようとする課題は、ウェブブラウザに対する操作(ブラウザ操作とも言う)を記録/再生する自動実行システムおよび方法を提供することである。また別の課題は、クライアント側にウェブブラウザ以外の特殊なソフトウェアをインストールする必要の無い自動実行システムおよび方法を提供することである。また別の課題は、再生時にネットワーク環境の変化に対応できる、ブラウザ操作自動実行システムおよび方法を提供することである。また別の課題は、ウェブ上のコンテンツを用いた自動実行型プレゼンテーションや、協調作業(コラボレーション)を実現する方法及びシステムを提供することである。また別の課題は、URLの遷移を記録できる、ブラウザ操作自動実行システムおよび方法を提供することである。また別の課題は、図形の描画機能の付加したブラウザ操作自動実行システムおよび方法を提供することである。また別の課題は、可変速再生や特定の操作のみを再生するなどの特殊記録・再生ができる、ブラウザ操作自動実行システムおよび方法を提供することである。また別の課題は、操作記録データサイズを従来技術より小さくする、ブラウザ操作自動実行システムおよび方法を提供することである。また別の課題は、記録したデータを専用のエディタを用いず容易に編集できる、ブラウザ操作自動実行システムおよび方法を提供することである。また別の課題は、ウェブブラウザに対する操作を記録・再生するためのイベントを発生させるメカニズムをHTMLファイルに挿入する方法を提供することである。

【0005】

【課題を解決するための手段】上記課題を解決するために、まずブラウザにイベント記録/再生プログラムを読み込み、コンテンツを表示するブラウザを起動する。次にコンテンツを表示するブラウザに、イベント記録または再生に必要な情報を収集するイベントハンドラの埋め込まれたページを読み込む。そしてコンテンツを表示するブラウザにおけるユーザのブラウザ操作に応じて、イベントハンドラに従い、イベントを発生させる。イベント記録/再生プログラムにより前記発生したイベントを記録し、記録したイベントをイベント記録/再生プログラムにより読み込む。読み込んだイベントに応じて、前記コンテンツを表示するブラウザを動作させる。

【0006】なおブラウザを起動する手段は、コンテンツを表示するブラウザを起動し、イベント記録/再生プログラムを読み込むブラウザを起動してもよい。またブラウザへイベント記録/再生プログラムを読み込む際、まずイベント記録プログラムだけを読み込み、再生時にイベント再生プログラムを読み込むようにしてもよい。本発明の本質を避けることなく適宜変更可能である。イベントハンドラの埋め込まれたページは、好ましくは、プロキシ・サーバでHTMLファイルにイベントハンドラを埋め込むことにより作成される。プロキシ・サーバは、専用のプロキシサーバでもよいし、通常のプロキシサーバ

にイベントハンドラの埋め込みを行わせてもよい。その他、イベントハンドラの埋め込みを行う別のプログラム(アプレット)を用意してもよい。何れにしても本発明の本質を外すことなく実施可能である。収集した情報は、好ましくは全てJavaアプレットにより記録される。再生は、JavaアプレットとJavaScriptによりウェブブラウザを制御することによって行なう。

【0007】図8にイベント記録のフローチャートを示す。ステップ810でイベント記録用のプログラムを起動し、ステップ820でイベントハンドラの埋め込まれたページの読み込みを行う。次にステップ830でユーザの操作に従い、イベントを発生させる。そしてステップ840で発生したイベント記録する。図9に記録したイベントを実行するフローチャートを示す。まずステップ910で記録されたイベント(イベントシーケンス)を読み込み、ステップ920で操作イベント(イベントシーケンス)の再生をブラウザに実行させる。

【0008】イベントを発生させるメカニズムを途中でHTMLファイルに挿入することにより、記録を考慮(レコーディング・アウェア)していない既存のウェブコンテンツを使った操作の記録・再生が可能となる。また記録したデータと、記録/再生を行なうソフトウェアの両者をウェブサーバ上に置くことにより、ウェブブラウザが利用できる環境であれば、記録/再生が可能である。ユーザの操作を記録するソフトウェア(以降、レコーダー)は全てJavaアプレットとJavaScriptで記述されるため、クライアント側では、ウェブブラウザ以外の特殊なソフトウェアを必要としない。つまりクライアント側にウェブブラウザ以外の特殊なソフトウェアをインストールする必要が無い。さらに、記録したデータには、コンテンツのURLとユーザが行なった操作に関する情報しか含んでおらず、従来技術に比べてデータのサイズが小さくなる。そして記録するデータをテキストファイルとすることで記録データの編集を容易にする。記録・再生する操作を選択したり、可変速度再生などの特殊操作も可能となる。また、複数のウェブブラウザの提供する機能に対して共通のコードを多く用いて、既存のウェブコンテンツへの任意の図形や文字の図形描画機能を提供する。このような自動実行機能や、図形描画機能により、ウェブ上のコンテンツを用いた自動実行型プレゼンテーションや、協調作業を実現する。

【0009】記録できる情報はウェブブラウザに対する操作(URLの遷移、ウェブのコンテンツが提供するテキスト・フィールド、ラジオ・ボタン、選択リストなどのフォームへの操作、ウェブのコンテンツが表示されているウィンドウへのウィンドウ操作、およびポインタ(マウス等)の移動)に対して行われる。さらにウェブのコンテンツが表示されているブラウザ・ウィンドウ上への任意の図形や文字の描画機能も提供し、この操作も記録する。

【0010】再生時は、記録した全ての操作イベントを再生することができる。この時、ブラウザの動作状況の確認を取りながら処理を進めることにより、ネットワーク環境の違いなどによるコンテンツの読み込み時間の変化に対応する。たとえばウェブブラウザを用いたキオスク(Kiosk)やホームバンキング(Home Banking)で操作上のトラブルがあった場合に、それを再生することにより問題の再現ができる。

【0011】さらに記録した操作イベントをネットワーク経由でリアルタイムに再生することにより、複数のノードでのウェブブラウザの同期を取ることが可能となる。たとえば先生が複数の生徒に対して、同一のコンテンツを表示操作したり、文字図形を描画して教育を行うなどに応用できる。さらに記録したコンテンツの再生速度、再生する操作イベントの種類などを制御することにより、より柔軟な再生が可能となる。例えば、URLの遷移のみを再生することにより、複数のウェブコンテンツのキャッシングを行なうことが可能となる。

【0012】

【発明の実施の形態】以下に本発明のブラウザ操作自動実行システムの実施例を説明する。

(1) 操作イベント取得機能(図1~2)

まず、記録の操作を行なうためのレコーダー・ウィンドウのページをウェブサーバから読み込む(図1の(1))。レコーダー・ウィンドウは、レコーダーのJavaアプレット、通信用のJavaアプレットを含んでいる。これらはイベント記録用のプログラムである。このページ中のJavaScriptのメソッドでコンテンツ・ウィンドウを開く(図1の(2))。コンテンツ・ウィンドウは、ユーザが見たいウェブページを表示するためのブラウザウィンドウである。従ってレコーダウィンドウとコンテンツウィンドウ2つのブラウザが起動している。なお最初にイベント記録用プログラムを読み込むブラウザ(レコーダウィンドウ)を起動し、コンテンツを表示するブラウザ(コンテンツウィンドウ)を起動するか、逆にコンテンツを表示するブラウザを起動し、イベント記録用プログラムを読み込むブラウザを起動するかは、実装上の自由選択であり、好ましいほうを選択してよい。コンテンツ・ウィンドウがプロキシ・サーバにコンテンツ読み込みを要求すると(図1の(3))、プロキシ・サーバは指定されたウェブサーバからコンテンツを読み込み(図1の(4))、読み込んだHTMLファイルに埋め込みエンジン(エンベディング・エンジン)でイベント・ハンドラを埋め込み、コンテンツ・ウィンドウに返す(図1の(5))。なおコンテンツウィンドウにはURL遷移を容易にできるようURLの入力フィールド表示しておいてもよい。

【0013】埋め込まれたイベント・ハンドラは、コンテンツ・ウィンドウで発生したイベントと再生に必要な情報を、レコーダー・ウィンドウに通知する。図2に、

HTMLファイルにイベント・ハンドラを埋め込んだ様子の例を示す。この例は、ウェブページの読み込みが完了したときに、そのイベントをレコーダー・ウィンドウに通知する例である。図2中、*****がつけられた行が埋め込まれた部分である(bodyタグの行に付いては、onLoad="notify()"の部分のみ)。図2のHTMLファイル中の"notify()"がイベントを通知するメソッドである。この"notify()"メソッドが呼ばれると、変数"msg"にイベント情報(時刻、イベントの種類、読み込んだコンテンツのURL、タイトル、および最終更新日)がセットされ、レコーダー・ウィンドウの"record()"メソッドに渡される。"recorder()"メソッドでレコーダーのJavaアプレットにイベント情報が渡されて記録される。

【0014】JavaScriptではこのように、ウィンドウのサイズ変化、ページの読み込み、ボタンなどのクリック、フォームの内容の変化、フォームのサブミット、マウスの移動などについて、それぞれonResize、onLoad、onClick、onChange、onSubmit、onMouseMoveなどでイベント・ハンドラを定義できる。これによって、再生時に必要な情報の発生をレコーダー・ウィンドウに通知することができる。スクロールに関する情報は、定期的にウェブブラウザのwindow.pageXOffsetプロパティとwindow.pageYOffsetプロパティを調べることで変化を通知することができる。

【0015】イベント・ハンドラの埋め込みをプロキシ・サーバで自動的に行なうことにより、ユーザがコンテンツに変更を加えることなく既存のコンテンツを利用できる。上記のようにJavaScriptでは、onXXXによってイベント・ハンドラを定義できるため、HTMLで書かれたオリジナルのコンテンツをパッケージングすることにより、適切な位置にそれぞれのイベント・ハンドラを挿入できる。また、埋め込みエンジンを持つプロキシ・サーバを置くことができない場合、埋め込みエンジンをもつアプレットをウェブサーバからダウンロードし、そのアプレットをローカル・プロキシ・サーバとして利用することにより、HTMLファイルにイベント・ハンドラを埋め込むことも可能である。その他、図1の(3)のリクエスト時にコンテンツウィンドウからのリクエストであることを判断することにより、汎用のプロキシサーバにHTMLファイルにイベント・ハンドラを埋め込む機能をもたせるようにしてもよい。

【0016】(2) 操作イベント記録機能(図3)

操作イベントの記録は、レコーダー・ウィンドウで走行するレコーダー・アプレットが行なう。レコーダー・ウィンドウでは、コンテンツ・ウィンドウのイベント・ハンドラから通知されてきた情報の取捨選択と整形を行ない、必要に応じてローカルファイルやネットワークに送る。それぞれについて以下に示す。

(a) ネットワークへの送信

アプレットの通信機能を用いてネットワークへ送る。ネ

ットワークの先で誰がどのように利用しているかはレコーダーは知る必要が無い。例えば、ウェブサーバ上においてあるプログラムが受け取ってそのままライブラリとしてウェブサーバ上に公開することも可能であるし、他のノードで走行するプレーヤーが受け取って、リアルタイムでのウェブブラウザの同期を行なうことも可能である。ただし、アプレットのセキュリティ上の制限があるため、アプレットは、自分がダウンロードされてきたサーバ（以降、自サーバ）としか通信コネクションを張ることができない。それ以外のノードと通信コネクションを張る場合には、自サーバ上に中継をしてくれるプログラムを置くか、もしくはアプレットに電子署名を付与することにより可能となる。

(b) ローカル・ファイルへの記録

アプレットに電子署名を付与することによりローカル・ファイルに記録することができる。

【0017】(3) 図形描画機能 (図4)

図形描画機能は、DynamicHTMLの機能を用いて実現する。本発明の図形描画機能はネットスケープコミュニケーター (Netscape Communicator 以下、NC) とインターネットエクスプローラ (Internet Explorer 以下、IE) で実現方法が異なる。これは、両者が採用しているDynamicHTMLに互換性がないためである。図4(a)にNCでの実現方法を示す。マウス・ポインタの動きにより、コンテンツ・ウィンドウ上に、小さな正方形のイメージを貼りつけたレイヤーを1つずつ作っていく。(なお色をつけたレイヤーを作成してもよい) レイヤー1上にイメージd1、レイヤー2上にイメージd2を作る。これを続けていくことにより、図4(b)のような図形が作成できる。マウス・ポインタの動きについては、マウス・イベントによりマウス・ポインタの動きを記録し、コンテンツ・ウィンドウ上のマウス・ポインタが通過した場所にレイヤを作り、マウス・ポインタのイメージを貼ることにより実現する。また、IEはレイヤーをサポートしていないため、マウス・ポインタが通過した場所にIEが提供するJavaScriptメソッド "insertAdjacentHTML()" によりイメージを貼りつけ、スタイル・シートの機能を利用して位置を調整する。

【0018】(4) 操作イベント読み込み機能 (図5)

操作イベントの読み込みは、ネットワークやローカル・ファイルから行なう。

(a) ネットワークからの読み込み

ウェブサーバ上で公開されている操作イベント・シーケンスは、通常のウェブコンテンツと同様にHTTPなどのプロトコルを用いてダウンロードすることができる。

(c) ローカル・ファイルからの読み込み

電子署名が付与されたアプレットにより、ローカル・ファイルから読み込むことができる。

【0019】(5) 操作イベント再生機能 (図5)

コンテンツ・ウィンドウの操作は、レコーダー・ウィン

ドウ上で走行するJavaアプレットとJavaScriptにより行われる。また、時間管理はJavaアプレットで行なう。

【0020】(5-1) コンテンツ・ウィンドウの操作
レコーダー・ウィンドウがコンテンツ・ウィンドウを開くときに (図1の(2)) コンテンツ・ウィンドウに名前を付与することで、レコーダー・ウィンドウからコンテンツ・ウィンドウをJavaScriptのメソッドを用いて制御できる。レコーダーアプレットは、再生時にレコーダー・ウィンドウのJavaScriptメソッドを呼び、そこからJavaScriptによってコンテンツ・ウィンドウを制御する。コンテンツ・ウィンドウに "win" という名前が付与されている場合、ウィンドウのサイズ変更、移動は、JavaScriptの win.resizeTo(x,y)、win.moveTo(x,y) メソッドで再生できる。また URLの遷移はコンテンツウィンドウの document.location.href プロパティに移動先のURLを代入することで再生できる。しかしブラウザのセキュリティのためレコーダウィンドウから直接コンテンツウィンドウの document.location.href プロパティを変更することはできない。そのため図1の埋め込みエンジンによってあらかじめコンテンツウィンドウに document.location.href プロパティを変更するメソッドを埋め込んでおきレコーダからそのメソッドを呼ぶことによってURL遷移を再生できる。このように、JavaScriptメソッドを利用することによって、操作イベントを再生することができる。

【0021】(5-2) 時間管理 (図6)

通常の再生時は、各イベント間の時間が記録時と同じようになるようにイベントの再生を行なう。ただし、ネットワークなどの状態によって、ウェブコンテンツの読み込み時間 (Loading Time) は大きく変化する恐れがある。ウェブコンテンツの読み込み時間は、図6(A)に示すように、load開始イベントからload終了イベントまでの間の時間 T_1 である。図6(B)に示すように、再生時にプレーヤーは、次ページの読み込みをコンテンツ・ウィンドウに指示したら、記録時の読み込み時間 T_3 に関係なくコンテンツ・ウィンドウからload終了イベントを受け取るまで待つ。また、記録時にload終了イベント前に行なわれた操作Aに関しては、読み込みが終了しないと操作できない危険性があるので、再生時にはコンテンツ・ウィンドウからload終了イベントを受け取った直後に行なう。通常速度再生では、読み込み時の表示時間 (Browsing Time) T_2 と同じ長さだけウェブページを表示する。つまり、図6において $T_4 = T_2$ とする。

【0022】(5-3) 特殊再生

(a) 再生情報の選択

再生時に、再生する情報種類を選択できる。これにより、URL遷移のみの再生したり、マウスの動作のみ再生しない、などの設定が可能となる。

(b) 再生開始ポイントの設定

任意のウェブページ読み込みイベントから再生させるこ

とが可能。

(c) 可変速再生

再生速度を変化させるために、図6の表示時間を調節する。2倍速再生の場合、読み込み時間>Loading Time)までの時間は変更せず、再生時の表示時間(BrowsingTime)を半分にする。つまり、 $T_4=T_2/2$ とする。

(d) リアルタイムでの再生時のQoS制御

リアルタイムで再生する場合、ネットワークの遅延などにより、記録側に対して再生側がどんどん遅れていく危険性がある。そこで、再生側では、再生する情報の種類ごとに遅延を認める閾値を決めておき、再生時に再生するか否かを決定することで、再生の遅延と再生の精度のバランスを調節できる。記録された情報をすべて再生しなくても、次のURLの読み込みで再生側はすべて同じ状態になる。

【0023】図7に、本発明のブラウザ操作自動実行を行うシステムのハードウェア構成例を示す。システム100は、中央処理装置(CPU)1とメモリ4とを含んでいる。CPU1とメモリ4は、バス2を介して、補助記憶装置としてのハードディスク装置13(またはMO、CD-ROM23、DVD等の記憶媒体駆動装置)とIDEコントローラ25を介して接続してある。同様にCPU1とメモリ4は、バス2を介して、補助記憶装置としてのハードディスク装置30(またはMO28、CD-ROM23、DVD等の記憶媒体駆動装置)とSCSIコントローラ27を介して接続してある。フロッピーディスク装置20はフロッピーディスクコントローラ19を介してバス2へ接続されている。

【0024】フロッピーディスク装置20には、フロッピーディスクが挿入され、このフロッピーディスク等やハードディスク装置13(またはMO、CD-ROM、DVD等の記憶媒体)、ROM14には、オペレーティングシステムと協働してCPU等に命令を与え、本発明を実施するためのコンピュータ・プログラムのコード若しくはデータを記録することができ、メモリ4にロードされることによって実行される。該コンピュータ・プログラムにはOS、ブラウザ(NC、IE)等のアプリケーション、その他のプログラムからなる。このコンピュータ・プログラムのコードは圧縮し、または、複数に分割して、複数の媒体に記録することもできる。

【0025】システム100は更に、ユーザ・インターフェース・ハードウェアを備え、入力をするためのポインティング・デバイス(マウス、ジョイスティック等)7またはキーボード6や、視覚データをユーザに提示するためのディスプレイ12を有することができる。また、パラレルポート16を介してプリンタを接続することや、シリアルポート15を介してモデムを接続することが可能である。このシステム100は、シリアルポート15およびモデムまたは通信アダプタ18(イーサネットやトークンリング・カード)等を介してネットワー

クに接続し、他のコンピュータ等と通信を行う。たとえばウェブサーバ、プロキシサーバ等とコミュニケーションする。またシリアルポート15若しくはパラレルポート16に、遠隔送受信機器を接続して、赤外線若しくは電波によりデータの送受信を行うことも可能である。なおウェブサーバ、プロキシサーバは上記システム100と同様の構成としても構わないし、専用のハードウェアを使用してよい。

【0026】スピーカ23は、オーディオ・コントローラ21によってD/A(デジタル/アナログ変換)変換された音声信号を、アンプ22を介して受領し、音声として出力する。また、オーディオ・コントローラ21は、マイクロフォン24から受領した音声情報をA/D(アナログ/デジタル)変換し、システム外部の音声情報をシステムにとり込むことを可能にしている。

【0027】このように、本発明のシステムは、通常のパーソナルコンピュータ(PC)やワークステーション、ノートブックPC、パームトップPC、ネットワークコンピュータ、コンピュータを内蔵したテレビ等の各種家電製品、通信機能を有するゲーム機、電話、FAX、携帯電話、PHS、電子手帳、等を含む通信機能有する通信端末、または、これらの組合せによって実施可能であることを容易に理解できるであろう。ただし、これらの構成要素は例示であり、その全ての構成要素が本発明の必須の構成要素となるわけではない。

【0028】

【発明の効果】本発明により、既存のウェブブラウザへの操作を記録することにより、ウェブコンテンツを用いた自動実行型のプレゼンテーション・パッケージを容易に作成することができる。記録を行なうレコーダは全てJavaアプレットとJavaScriptで記述されるため、クライアント側では、ウェブブラウザ以外の特殊なソフトウェアを必要としない。再生時にウェブブラウザの動作状況を確認しながら処理を進めることにより、ネットワーク環境の違いなどによるコンテンツの読み込み時間の変化などに対応できる。ウェブのコンテンツを表示するウィンドウ上への図形描画機能を提供することにより、より柔軟なプレゼンテーションが可能となる。記録した操作イベントをネットワーク経由でリアルタイムに再生することにより、複数のノードでのウェブブラウザの同期を取ることが可能となる。記録したコンテンツの再生速度、再生する操作イベントの種類などを制御することにより、より柔軟な再生が可能となる。例えば、URLの遷移のみを再生することにより、複数のウェブコンテンツのキャッシングを行なうことが可能となる。

【図面の簡単な説明】

【図1】操作イベントの取得方法を示す図である。

【図2】HTMLへのイベントハンドラの埋め込み例を示す図である。

【図3】操作イベント記録機能を説明する図である。

【図4】図形描画機能を説明する図である。

【図5】操作イベント読み込み／再生機能を説明する図である。

【図6】再生時の時間管理を説明する図である。

【図7】本発明の自動実行システムに使用されるハード

ウェアの1実施例を示す図である。

【図8】本発明のイベント記録のフローチャートである。

【図9】本発明の記録イベント実行フローチャートである。

【図2】

```

<html>
<head>
* <script language="JavaScript">
* function notify(){
*   var now = new date();
*   var msg = now.getTime()+"load,"+document.location.href+ ","+
*             document.title+"."+document.lastModified;
*   opener.record(msg);
* }
* </script>
</head>
* <body onLoad="notify()">

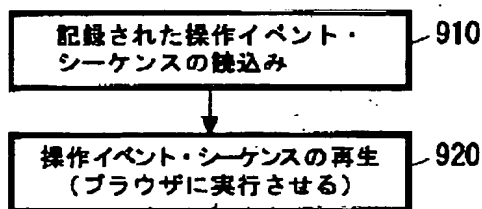
  original contents

</body>
</html>

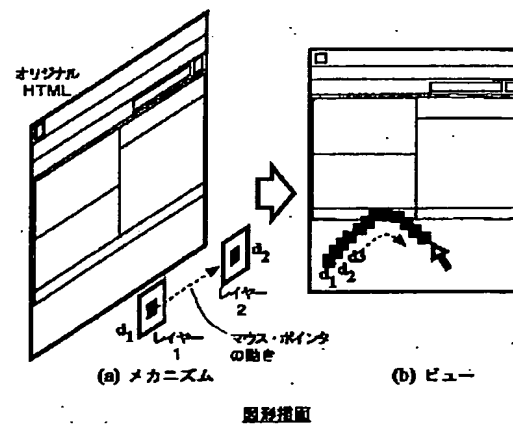
```

HTMLへのイベント・ハンドラの埋込み例

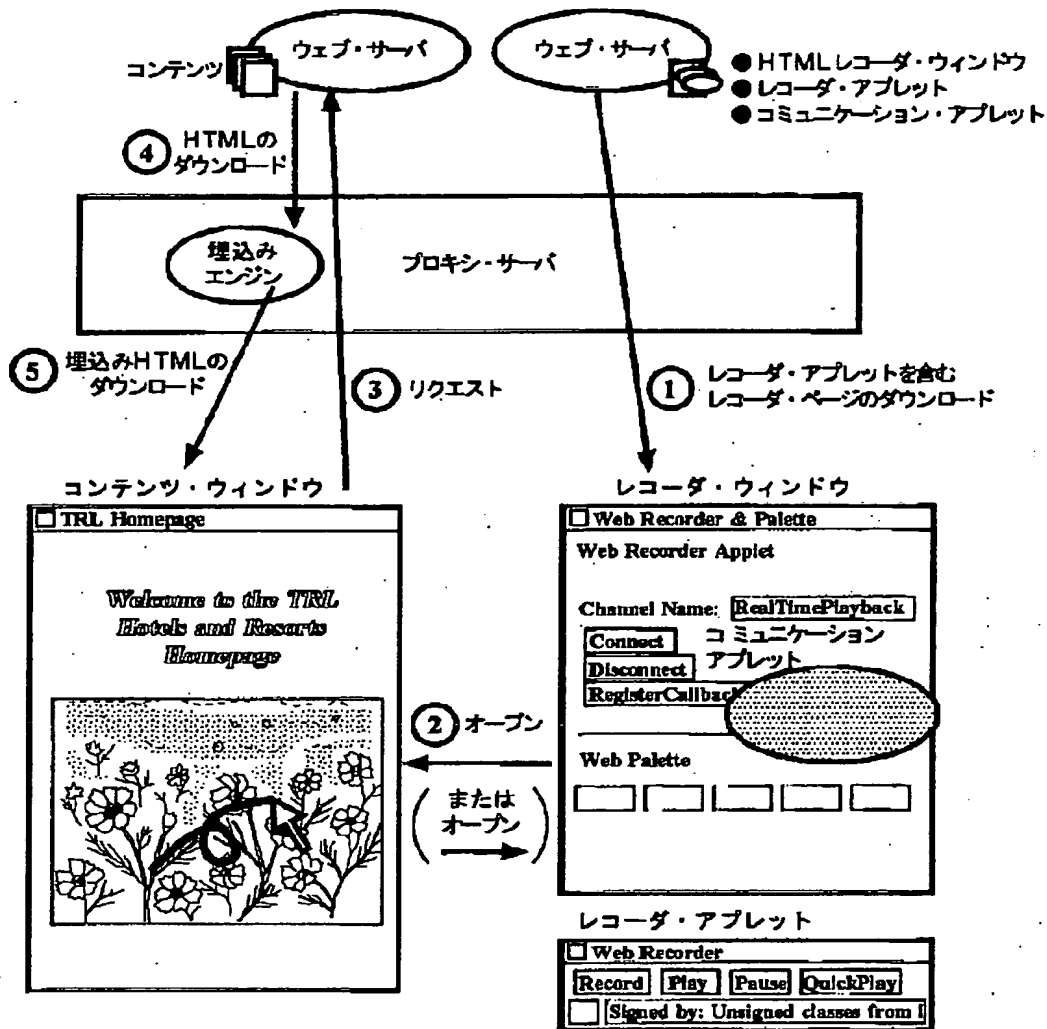
【図9】



【図4】

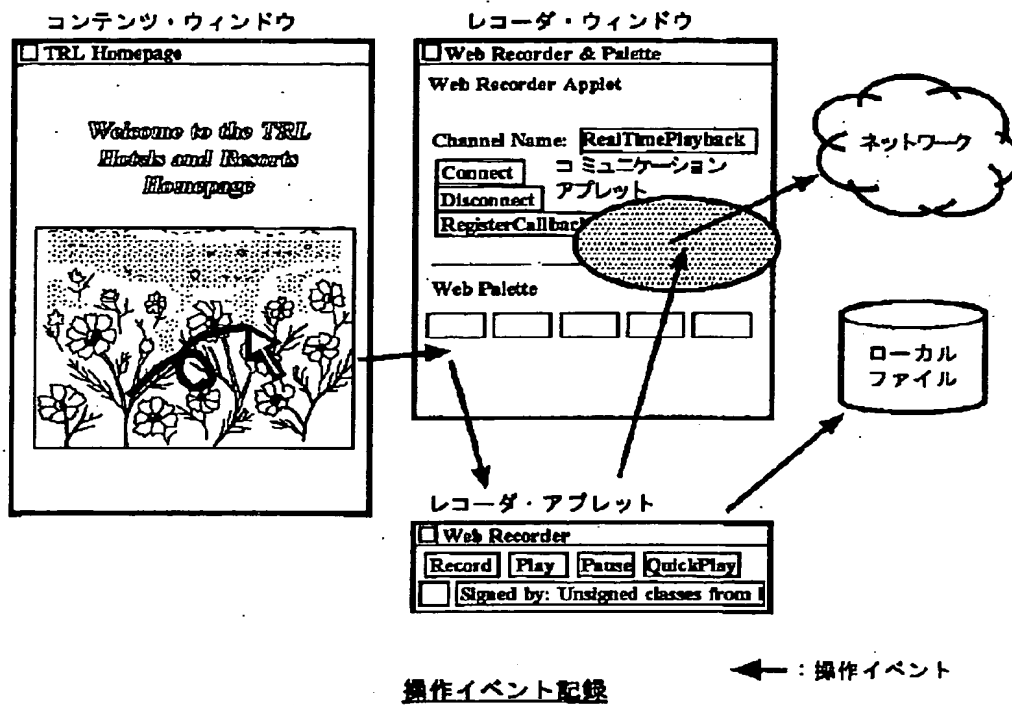


【図1】

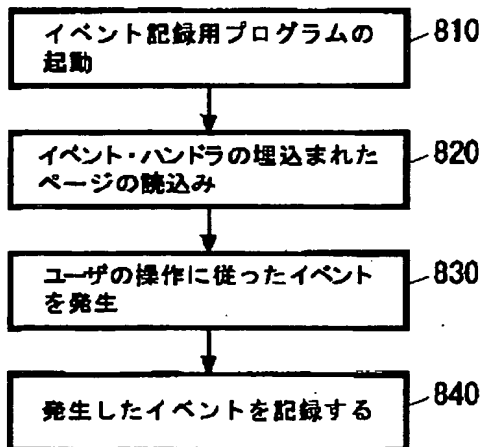


操作イベントの取得

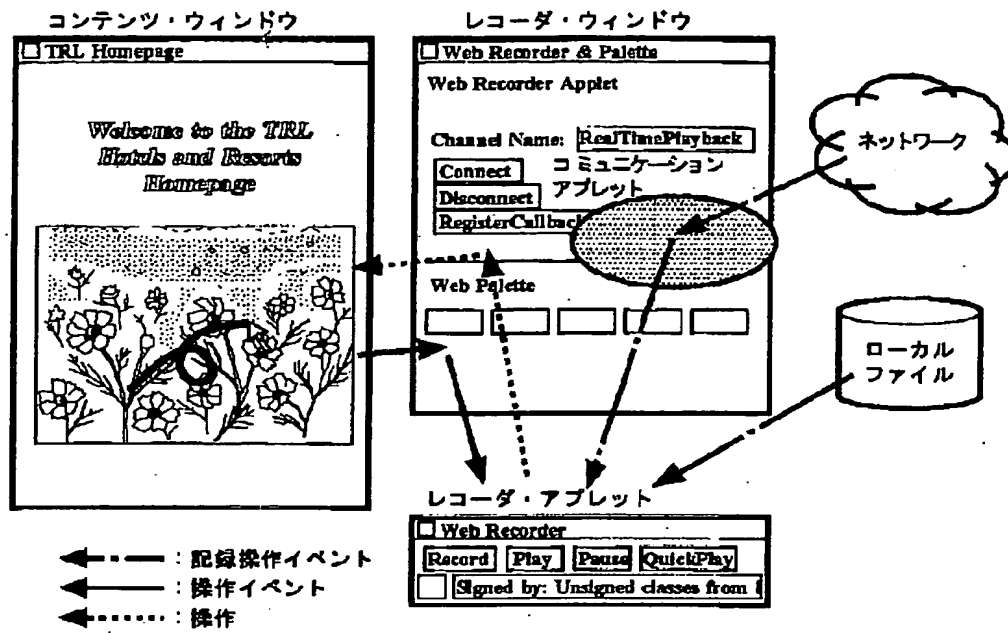
【図 3】



【図 8】

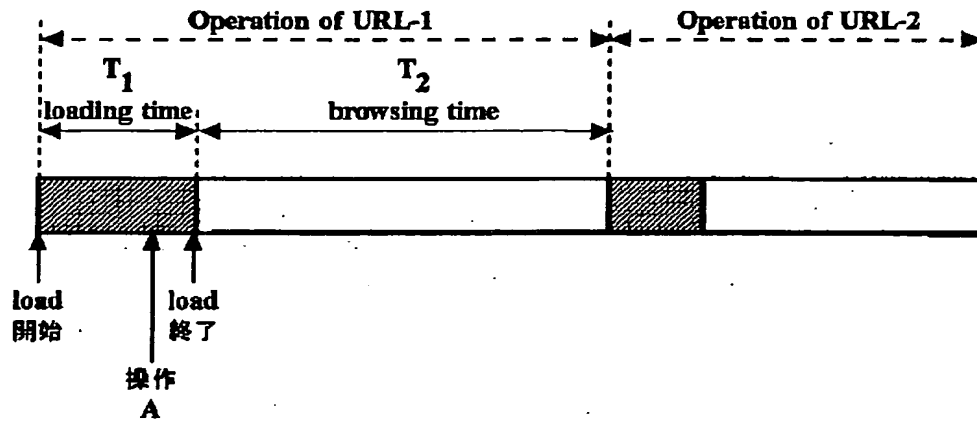


【図 5】

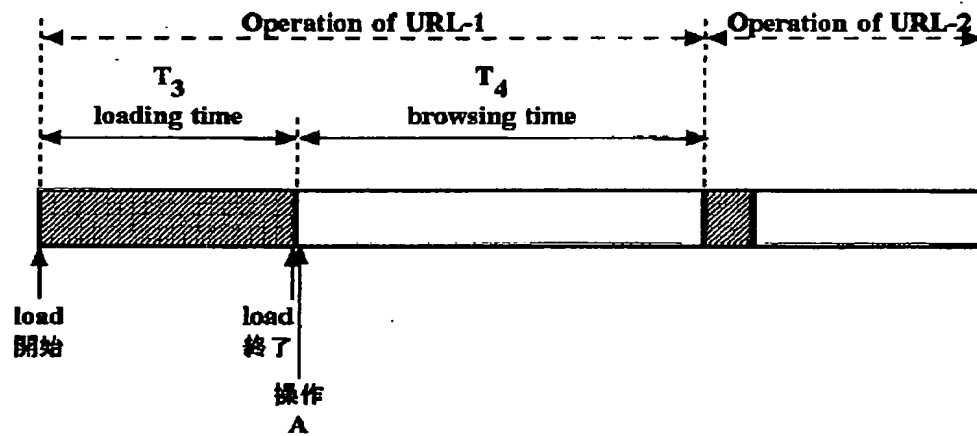


【図 6】

(A) 記録

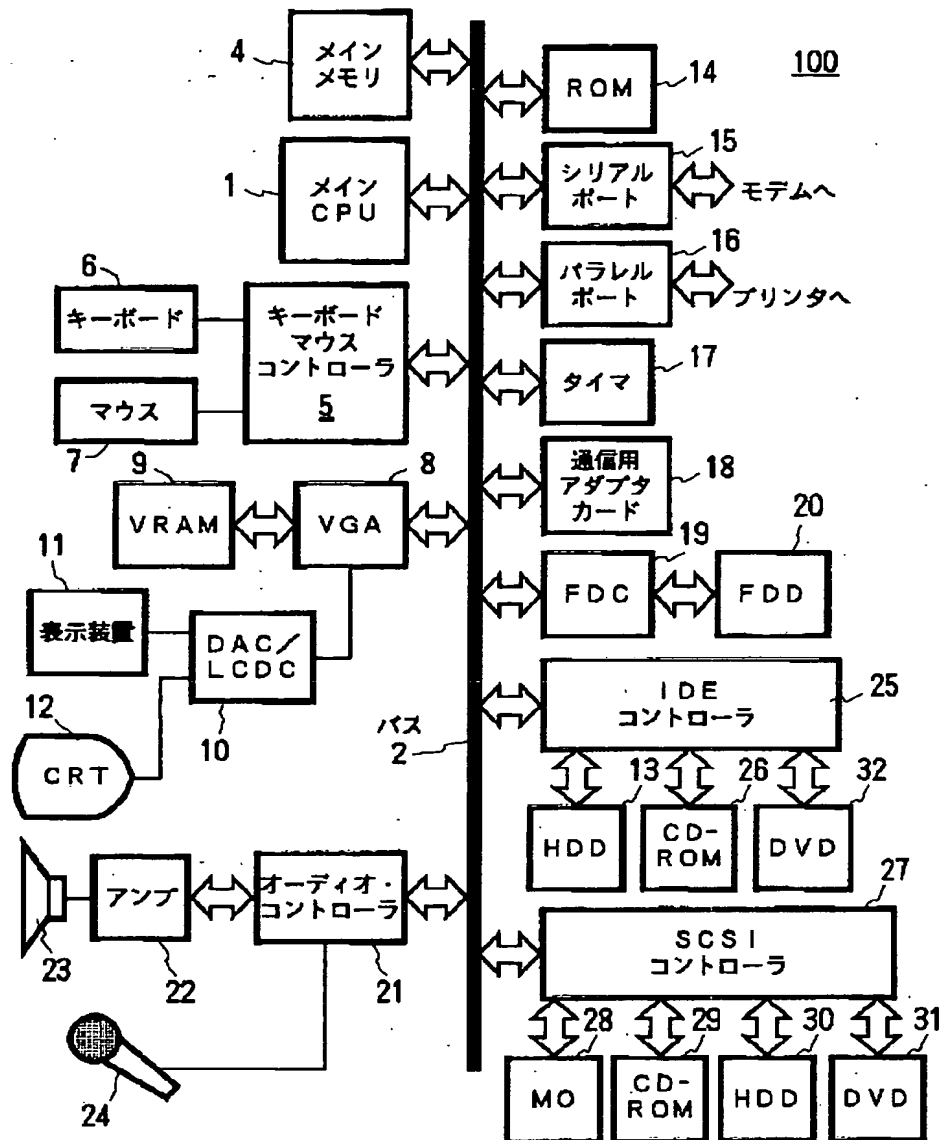


(B) 再生



時間管理

【図7】



フロントページの続き

(72)発明者 安藤 史郎
 神奈川県大和市下鶴間1623番地14 日本
 アイ・ビー・エム株式会社 東京基礎研
 究所内

(72)発明者 中島 周
 神奈川県大和市下鶴間1623番地14 日本
 アイ・ビー・エム株式会社 東京基礎研
 究所内

(56)参考文献 特開 平5-67150 (JP, A)
特開 平5-242166 (JP, A)
「はじめてのJavaScript第
3回 イベントとイベントハンドラ」C
MAGAZINE, Vol. 8, N
o. 12, 1996, p. 96-101 (平8-12
-1)

(58)調査した分野(Int.Cl.⁷, DB名)
G06F 17/30
G06F 11/34
JICSTファイル(JOIS)